

**Vysoká škola báňská – Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky**

**Aplikace platformy Raspberry Pi pro měření ve velkých  
výškách pomocí balónu**

**Application of Raspberry Pi platform for High Altitude  
Balloon Measurement**

**2014**

**Radek Škarvada**

VŠB - Technická univerzita Ostrava  
Fakulta elektrotechniky a informatiky  
Katedra informatiky

## Zadání bakalářské práce

Student:

**Radek Škarvada**

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Aplikace platformy Raspberry Pi pro měření ve velkých výškách pomocí balónu

Application of Raspberry Pi Platform for High Altitude Balloon Measurement

Zásady pro vypracování:

Cílem práce je návrh a implementace komplexního řešení měřicí výbavy pro měření ve velkých výškách pomocí balónu. Výsledné řešení nabídne unifikovanou platformu pro dané použití. Jako základ bude využita platforma Raspberry Pi s OS GNU/Linux.

1. Seznamte se důkladně s platformou Raspberry Pi. Prostudujte existující projekty pro měření ve velkých výškách. Zaměřte se na především na pořizování obrazu, použitá čidla a řešení přenosu získaných dat.
2. Definujte požadavky pro návrh řešení s ohledem na možnosti dané použitou platformou.
3. Navrhněte řešení sestávající se z existujících komponent. K řešení implementujte řídící aplikaci.
4. Řešení otestujte v simulovaných testovacích podmínkách a řádně zdokumentujte.

Seznam doporučené odborné literatury:

- [1] EBEN, Upton; HALFACREE, Gareth. Raspberry Pi User Guide. Wiley, United Kingdom, 2012. ISBN 978-1-118-46446-5
- [2] KADLEC, Václav. Učíme se programovat v jazyce C. Brno, CP Books, 2005. ISBN 80-7226-715-9
- [3] HERTZOG, Raphaël; MAS, Roland. The Debian Administrator's Handbook. 2012. ISBN: 979-10-91414-00-5

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Daniel Stříbný**

Datum zadání: 01.09.2013

Datum odevzdání: 07.05.2014



doc. Dr. Ing. Eduard Sojka  
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.  
děkan fakulty

## **Prohlášení studenta**

Prohlašuji, že jsem tuto bakalářskou/diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne: 17. července 2014

Handwritten signature in blue ink, appearing to read "Tomáš Barch".

podpis studenta

## **Poděkování**

Rád bych poděkoval mému vedoucímu panu Ing. Danielovi Stříbnému za odbornou pomoc a konzultace při zpracování této bakalářské práce.

## **Abstrakt**

Tato bakalářská práce se zabývá návrhem měřicí výbavy pro měření ve velkých výškách nad zemí pomocí balónu a s využitím platformy Raspberry Pi. Jako základ bude použit kompaktní minipočítač Raspberry Pi s operačním systémem Linux, který bude řídit komunikaci mezi jednotlivými moduly. Počítač Raspberry Pi bude spolupracovat s navrženými moduly jako je kamera pro pořízení obrazu, GPS modul pro určení geografické polohy balónu, nebo teploměr pro měření okolní teploty.

## **Klíčová slova**

balón; Raspberry Pi; GPS přijímač; teploměr; Linux; Raspbian

## **Abstract**

This bachelor thesis describes the design of the measuring equipment for measurement at high altitudes above the ground using a balloon and using the platform Raspberry Pi. Will be used as the basis of a compact mini computer Raspberry Pi with Linux operating system, which will manage communication between components. The computer Raspberry Pi will work with modules designed as a camera for image acquisition, GPS module to determine the geographic location of the balloon, or a thermometer to measure ambient temperature.

## **Key words**

balloon; Raspberry Pi; GPS receiver; thermometer; Linux; Raspbian

## Seznam použitých symbolů a zkratek

Zkratka	Význam
<b>HAB</b>	Balón pro měření ve velkých výškách(High altitude balloon)
<b>USB</b>	Univerzální sériová sběrnice(Universal serial bus)
<b>GPS</b>	Globální polohovací systém(Global position system)
<b>C/C++</b>	Programovací jazyk
<b>ARM</b>	Architektura mobilních procesorů(Advanced RISC Machine)
<b>Video Core</b>	Mobilní architektura procesoru
<b>SD</b>	Paměťová karta(Secure digital)
<b>MMC</b>	Paměťová karta(MultiMediaCard)
<b>RCA</b>	Konektor pro připojení videa/audia(Radio Corporation of America)
<b>HDMI</b>	Audio/video připojení(High definition Multi-Media Interface)
<b>GPIO</b>	Vstupně/výstupní konektor(General-purpose input/output)
<b>BSD</b>	Svobodná licence(Berkeley Software Distribution)
<b>Clk</b>	Hodinový signál,střídající se logické 1 a 0(clock)
<b>UART</b>	Asynchronní sériové rozhraní
<b>SPI</b>	Sériové periferní rozhraní(Serial Peripheral Interface)
<b>CSI</b>	Rozhraní pro připojení kamery k Raspberry Pi(Camera serial interface)
<b>b</b>	Nejmenší jednotka informace(bit)
<b>B</b>	Jednotka množství dat(Byte = 8 bitů)
<b>W</b>	Jednotka výkonu/příkonu(Watt)
<b>A</b>	Jednotka elektrického proudu(Ampér)
<b>V</b>	Jednotka elektrického napětí(Volt)

# Obsah

1	Měření ve velkých výškách pomocí balónů .....	- 11 -
1.1	Rozdělení balónů .....	- 11 -
1.1.1	Meteorologické měřicí balóny .....	- 11 -
1.1.2	Experimentální měřicí sondy .....	- 11 -
1.1.3	Stratosféra .....	- 12 -
1.2	Princip vypouštění balónů .....	- 12 -
1.3	Raspberry Pi .....	- 13 -
1.3.1	Hardware .....	- 13 -
1.3.2	Operační systém .....	- 15 -
1.3.3	GPIO porty .....	- 16 -
1.4	Měřicí výbava .....	- 20 -
1.4.1	Payload box .....	- 20 -
1.4.2	Komponenty .....	- 21 -
1.4.3	GPS .....	- 21 -
2	Požadavky pro návrh řešení .....	- 23 -
2.1	Použití Raspberry Pi .....	- 23 -
2.2	Měřicí výbava pro Raspberry Pi .....	- 24 -
3	Návrh konkrétního řešení .....	- 26 -
3.1	Programovací jazyk .....	- 26 -
3.1.1	C pro Raspberry Pi .....	- 26 -
3.2	Použité komponenty .....	- 26 -
3.2.1	Raspberry Pi Camera Board .....	- 26 -
3.2.2	GPS modul RoyalTek RGM-2000 .....	- 29 -
3.2.3	Teploměr Dallas DS18B20 .....	- 30 -
3.2.4	Celkové zapojení .....	- 31 -
3.3	Řídící aplikace .....	- 33 -
3.3.1	Pořízení obrazu .....	- 36 -
3.3.2	Získání GPS souřadnic .....	- 37 -
3.3.3	Čtení teploty .....	- 38 -



4	Testování .....	- 40 -
4.1	Napájení .....	- 40 -
4.2	Testování v simulovaných podmínkách .....	- 42 -
	Závěr .....	- 45 -
	Použitá literatura .....	- 46 -
	Seznam příloh.....	- 48 -

# Úvod

Minipočítač Raspberry Pi se vzhledem k jeho rozměrům, možnosti připojení externích komponent a spotřebě energie hodí na nejrůznější použití, zpracování dat a řízení procesů.

V mé bakalářské práci navrhuji použití tohoto minipočítače pro aplikaci zaměřenou na měření ve velkých výškách za pomoci balónu, který by vynesl celý měřicí box včetně počítače Raspberry Pi se všemi přídatnými moduly do stratosféry do výšky cca 40km. Zde se již nachází prostředí s nízkým tlakem a průměrnou teplotou přibližně  $-40\text{ }^{\circ}\text{C}$  a je potřeba těmto podmínkám přizpůsobit návrh měřicí výbavy.

V první části popisují problematiku vypouštění meteorologických balónů s měřicí výbavou do stratosféry. Co vše je zapotřebí měřit, jaké komponenty se využívají pro získávání cenných dat ve velké výšce. Dále popisují výhody a nevýhody použití právě platformy Raspberry Pi pro tento návrh, srovnávám tuto platformu s konkurenčními produkty. Porovnávám možnosti použitých čidel a modulů s primárním zaměřením pro pořízení obrazu, zjištění aktuální polohy balónu pomocí GPS a získání naměřených dat z použitých čidel.

Ve druhé části se věnuji definování požadavků pro návrh řešení obsahující přehled vhodných komponent s ohledem na možnosti platformy Raspberry Pi. Použití jednotlivých modulů pro pořízení obrazu, měření teploty a získávání informací o aktuální zeměpisné poloze. Dále popisují co by měl obsahovat výsledný měřicí box ve kterém bude celá měřicí výbava, řeším problémy spojené s výběrem nejvhodnějších komponent do prostředí velkých výšek a pro spojení právě s mobilní platformou Raspberry Pi.

V další části řeším konkrétní návrh zapojení skládajícího se z již existujících komponent, které jsem navrhnul pro použití v této práci. Srovnávám konkrétní existující moduly a popisují výhody zvolených komponent. Dále se zde věnuji návrhu řídicí aplikace v programovacím jazyce C pro komunikaci s externími moduly, obsahující záznam fotografií a videa pomocí kamery, čtení aktuální polohy pomocí GPS a záznam teploty. Programovací jazyk C je vhodný právě pro použití řízení jednotlivých modulů s použitím minipočítače Raspberry Pi s operačním systémem Linux.

V poslední části se věnuji testování celkového měřicího vybavení s minipočítačem Raspberry Pi. Popisují zde možné vhodné napájení měřicího boxu, včetně měření spotřeby jednotlivých komponent i celkového zapojení. Dále jsou zde zdokumentovány testy v simulovaných podmínkách, jako jsou měření za nízkých teplotních podmínek, záznam snímků a videa v terénu a dekodování GPS souřadnic na zvolené trase.

# 1 Měření ve velkých výškách pomocí balónů

## 1.1 Rozdělení balónů

### 1.1.1 Meteorologické měřicí balóny

Jedná se o bezpilotní balóny obvykle plněné héliem, nebo vodíkem, které vynesou box s měřicí výbavou do výšky cca 20-40 km nad zemský povrch. V této výšce se nachází stratosféra s teplotou od -15 až do cca -70°C. Nejběžnějším typem těchto balónů vypouštěným do velkých výšek jsou meteorologické balóny. Tyto balóny vynášejí do stratosféry speciálně navržené sondy obsahující čidla pro měření tlaku, teploty, rychlosti proudění větru, radioaktivity a dalších veličin potřebných pro výpočet předpovědi počasí. Meteorologické balóny jsou vypouštěny několikrát denně, měřicí box je ve většině případů zkonstruován jako jednorázový a po odpojení od balónu a následném pádu se již tyto boxy nedají použít a meteorologické ústavy se o polohu dopadu měřicí sondy dále většinou nezajímají.

### 1.1.2 Experimentální měřicí sondy

Stejně jako meteorologické balóny vynášejí různé boxy(sondy) do výšky stratosféry, jen nejsou přímo navrženy a sestaveny pro přesné měření veličin jen pro meteorologické potřeby. Zde se experimentuje s různými dalšími čidly pro měření ve velkých výškách, v měřicích boxech se využívají GPS moduly pro přesné geografické zaměření polohy sondy. A hlavně tyto balóny s sebou nesou ve většině případů také fotoaparát nebo kameru pro zachycení snímků ze stratosféry nebo tzv. blízkého vesmíru, tyto snímky zobrazují pohled vysoko nad zemským povrchem a jde na nich vidět samotné zakřivení Země, pro ilustraci přikládám obrázek pořízený balónem ze stratosféry obr.1.1. [2]



Obrázek 1.1: *Obrázek pořízený z balónu [2]*

### 1.1.3 Stratosféra

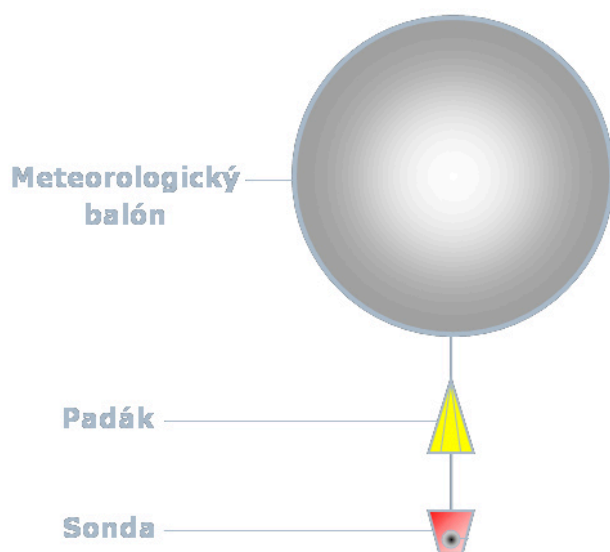
Balóny s měřicím boxem vystoupají do úrovně stratosféry, zde se nacházejí odlišné klimatické podmínky.

Stratosféra je dlouhá cca 50 km. Tato vrstva má 19 procent plynů v atmosféře, ale velmi málo vodní páry. V této oblasti se teplota zvyšuje s výškou. Teplo se tvoří v procesu tvorby ozonu a toto teplo je zodpovědné za zvýšení teploty od průměrné  $-51^{\circ}\text{C}$  od tropopauzy až na nejvyšší do  $15^{\circ}\text{C}$ , při horní vrstvě stratosféry. [1]

## 1.2 Princip vypouštění balónů

Zařízení pro vypouštění experimentální sondy by se mělo skládat z těchto částí (viz obr. 1.2).

1. meteorologický balón
2. padák (zmírňující dopad)
3. sonda (obsahující veškerou techniku)



Obrázek 1.2: balón se sondou a padákem [4]

Meteorologický balón naplněný héliem je schopný vynést sondu až do výšky přes 30 km. Hélium má menší hustotu než atmosférický vzduch, přitom čím výš balón vystoupá, tím menší tlak na něj bude působit. Tyto dvě skutečnosti způsobí pomalé zvětšování balónu, který ve výšce okolo 30 km praskne. Například meteorologický balón vážící 1,5 kg a schopný unést náklad (do 1 kg), má při nafouknutí ve svém průměru 185 cm. Po vystoupení do své maximální výšky 34,2 km jeho průměr dosahuje již 944 cm. Tlak v této výšce (6,3 hPa) pak zaručeně způsobí prasknutí takto zvětšeného balónu. Po několika málo vteřinách beztlaku začne celý náklad padat k zemi. Se zrychlením se napne padák, který byl původně zavěšený pod balónem. Ten zajistí, aby celá sonda dopadla zpátky na zem s přiměřenou rychlostí, zaručující nepoškození jejího obsahu. Samotný povedený let je jen jedna část úspěchu. Tou druhou je správné fungování obsahu sondy. Tím se dostáváme k tomu nejdůležitějšímu, a to je obsah boxu. Ten by měl obsahovat řídicí počítač, čidla pro měření, gps přijímač a čidla pro měření. Vzhledem k tomu, že je ve stratosféře teplota pohybující se okolo -40 °C (což box musí přežít) a mohou v něm vznikat vzdušné proudy o rychlosti několika set kilometrů v hodině, je nutné, aby technika a především její baterie zůstaly v teple. Proto je nutné, aby byl box perfektně utěsněný. [4]

Poslední důležitou záležitostí je váha boxu. Nesmí být moc těžký, a to jak kvůli dosažení dostatečné výšky, tak kvůli splnění váhových podmínek. Faktická doba setrvání balónu na úrovni, ve které létají dopravní letadla je v řádu vteřin, letadla mají právě pro případ kolize povinně dva motory, navíc za celou historii letectví není znám případ kolize letadla s meteorologickým balónem. Kromě toho materiál balónu a hmotnost samotné kapsle by v případě kolize s letadlem zajistily rozmetání veškerého našeho zařízení. [4]

## 1.3 Raspberry Pi

V této bakalářské práci se věnuji návrhu měřicího vybavení pro balón s použitím malého jednodeskového minipočítače jménem Raspberry Pi, který bude řídit veškerou komunikaci mezi všemi komponentami obsaženými v měřicím boxe(sondě).

### 1.3.1 Hardware

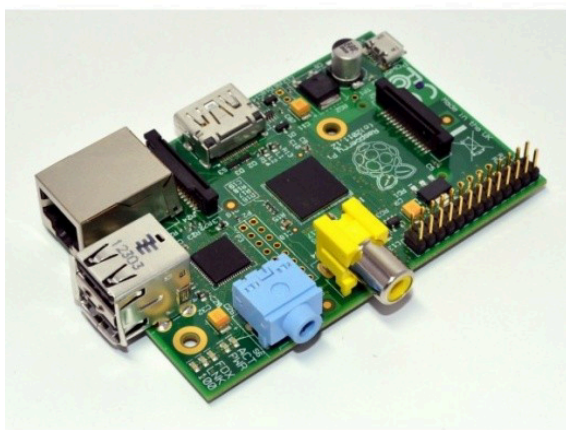
Raspberry Pi je malý počítač o velikosti zhruba platební karty. S cílem podpořit studenty při výuce informatiky ho vyvíjí britská nadace Raspberry Pi Foundation.

Tento počítač se vyrábí ve dvou verzích, které se liší především použitím jiné velikosti operační paměti a výbavou portů. Model B obsahuje oproti modelu A 512Mb operační paměti, zatímco model A má jen 256Mb. Dále obsahuje model B navíc síťový adaptér s konektorem RJ45 a dvěma USB porty, model A má pouze jeden. Ostatní použitý hardware mají oba modely

shodný. Na obrázku 1.3 je fotografie modelu B, který byl použit pro návrh v této bakalářské práci. [7]

Hardware:

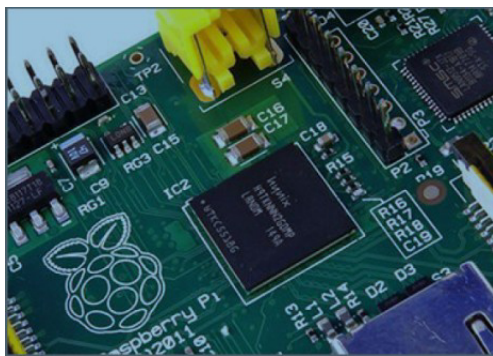
- procesor ARM s taktem 700 Mhz
- grafický procesor VideoCore 4
- 256, nebo 512Mb operační paměti
- jeden, nebo 2 USB porty
- slot SD nebo MMC kartu
- zvukový výstup jack 3,5mm konektor
- obrazový výstup HDMI a RCA composit
- 8 GPIO portů, SPI sběrnice



Obrázek 1.3: *obr. 1.3 Raspberry Pi model B 1*

#### Architektura ARM

Srdcem minipočítače Raspberry pi je multimediální procesor SoC Broadcom BCM28354. To znamená že je vše ukryto v jednom čipu, 256MB operační paměti, grafická jednotka, audio, komunikační hardware. Tento procesor používá architekturu ISA známou jako ARM a pracuje s instrukční sadou ARMv6. Celý čip procesoru je vidět na obrázku 1.4 [7]



Obrázek 1.4: *obr.1.4 Soc Broadcom 1*

### 1.3.2 Operační systém

Raspberry Pi používá ke své funkci jako operační systémy primárně systém, založený na Linuxovém jádře (operační systémy Linux).

Na rozdíl od systému Windows, nebo OS X, Linux je open source, lze stáhnout jeho zdrojový kód pro celý operační systém a upravit jej dle potřeby. Několik systémů Linux bylo právě upraveno pro procesor Raspberry Pi Broadcom BCM28354, zahrnující linuxové distribuce jako je Debian, Fedora, ArchLinux. Všechny tyto linuxové systémy jsou také navzájem kompatibilní, to znamená, že program napsaný v Debian Linuxu bude fungovat i na Arch Linuxu apod. Zde jsou nejpoužívanější a nevhodnější operační systémy pro Raspberry Pi.

Raspbian:

Jako nejpoužívanější a také doporučovaný systém pro tento minipočítač se používá Linuxový systém s názvem Raspbian. Byl vytvořen speciálně pro minipočítač Raspberry Pi. Raspbian je založený na svobodném operačním systému Debian (Linuxový systém) a je optimalizovaný pro hardware, který je použit na Raspberry Pi. Pro jeho instalaci na minipočítač je zapotřebí minimálně 2GB volné paměti na paměťové kartě a doporučeno je mít volné 4GB paměti. Operační systém Raspbian má již v základu předinstalovaných několik užitečných aplikací, například internetový prohlížeč Midori a různé programovací nástroje.

Debian Linux upravený pro Raspberry Pi (Raspbian) obsahuje s hlediska uživatelského rozhraní pracovní plochu nazvanou Lightweight X11 Desktop Environment, toto prostředí je navrženo, tak aby bylo uživatelsky jednoduché a podobné prostředí pracovní plochy ze systému Windows. [7]

Další operační systémy:

Na Raspberry pi je možno nainstalovat i jiné operační systémy, které jsou založeny na Linuxovém jádře. Například oficiálně podporované jsou kromě Raspbianu ještě ArchLinux a Pidora, které jsou také založeny na Linuxových distribucích.

Jelikož je ale pro Raspberry Pi doporučen a také nejlépe vyladěný pro jeho hardware operační systém Raspbian, budu se dále zabývat jen tímto systémem, který jsem také použil při návrhu této práce.

### 1.3.3 GPIO porty

Kromě klasického USB portu je minipočítač vybaven pro připojení různých externích zařízení (jako je v mé práci například teploměr, nebo GPS přijímač). Počítač Raspberry Pi je také vybaven GPIO porty, to jsou tzv. univerzální vstupně výstupní porty, které se dají využít pro obousměrnou komunikaci mezi Raspberry Pi a externími moduly. GPIO konektor [6] obsahuje 2x 13 pinů, z toho tvoří 8x univerzální GPIO porty a dále zahrnuje sběrnice SPI, I2C a linku pro sériový přenos dat UART a nakonec také obsahuje napájecí piny a to s napětím 5V a 3,3V, což je vhodné pro většinu přídavných elektronických modulů a čidel. Z Raspberry Pi modelu B je možno odebírat maximální proud 500mA. Celé rozložení konektoru je zobrazeno na obrázku 1.5. [7]





Obrázek 1.5: GPIO konektor na Raspberry Pi

I když GPIO port poskytuje napájecí pin 5V, které je odebíráno z napájecího micro usb konektoru, vnitřní architektura Raspberry Pi pracuje s napětím 3,3V. Pokud chceme tedy k Raspberry pi připojit externí komponenty musíme se ujistit, že pro komunikaci používá právě 3,3V logiku. Pro univerzální použití poskytuje GPIO piny 11,12,13,15,16,18,22. [14]

Přístup k portům v systému:

Existují 2 základní možnosti jak přistupovat na GPIO porty na Raspberry Pi.

První z nich je takový, že můžeme manipulovat přímo s registry v Raspberry Pi, takto se komunikuje běžně při použití různých mikroprocesorů bez OS (operačního systému). Výhodou tohoto přímého přístupu je, že objdeme operační systém, v případě Raspberry Pi je to Linux a tím pádem se hodnota pinů dá rychleji změnit. Naopak nevýhodou tohoto přístupu je bezpečnost, jelikož když se 2 procesy (instance programů) snaží zároveň přistupovat k fyzickému registru GPIO pinu tak může dojít ke konfliktu a následně ke ztrátě cenných dat. [12]

Druhý způsob je bezpečnější a pro počítač Raspberry Pi rozumnější, jelikož je na něm nainstalován operační systém Raspbian (upravená verze systému Linux). Jedná se o přístup přes samotný systém Linux, tento operační systém má již v sobě vestavěný ovladač pro bezpečný přístup k GPIO portům. Princip je takový, že operační systém považuje každou vlastnost u všech pinů jako jeden příslušný soubor. Jedinou nevýhodou tohoto přístupu přes operační systém je pomalejší přepínání stavů portů, ale na druhou stranu je tento přístup opravdu bezpečnější a to je pro bezproblémový chod řídicí aplikace pro mé využití důležitější. [12]

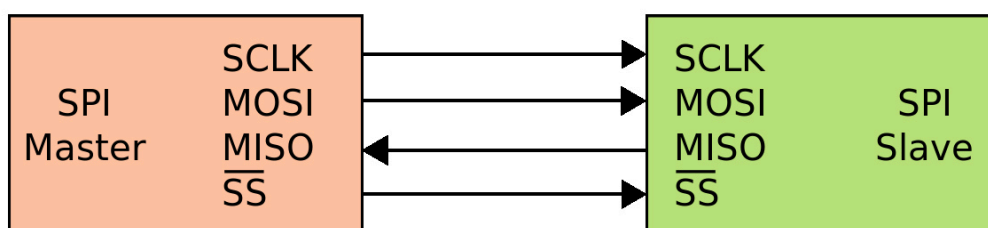
Nyní se budu věnovat sběrnicím a rozhraním, které jsou již integrovány a připraveny na GPIO portu na Raspberry Pi a nabízí různé možnosti pro komunikaci s externími komponenty.

SPI:

Jedná se o sériové periferní rozhraní. Používá se ke komunikaci mezi řídicími mikroprocesory (v našem případě počítač Raspberry Pi) a ostatními integrovanými obvody. Pomocí speciálních vodičů probíhá adresace, kde se aktivuje při logické nule příjem a vyslání dat z nebo do daného připojeného zařízení. Komunikace probíhá na společné sběrnici, to znamená že jsou všechny zařízení propojeny s Raspberry Pi na stejné lince. Zařízení jsou rozděleny na hlavní(master) a ostatní připojené (slave). [7]

Na rozdíl od komunikace I2C a UART jde o čtyřvodičovou sběrnici, umožňující komunikaci s více než jedním cílovým zařízením.

Master, to může být v našem případě počítač Raspberry Pi generuje hodinový signál (clk) a řídí celou komunikaci (určuje s jakým zařízením bude komunikovat, jeli jich na sběrnice připojeno více). Slave, to znamená připojené periferní zařízení, pokud je aktivován Masterem tak vysílá data podle hodinového signálu (clk).



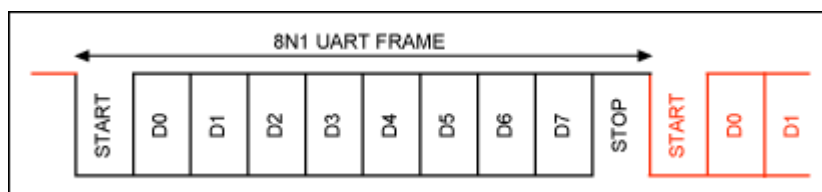
Obrázek 1.6: SPI komunikace [7]

Na obrázku 1.6 je zobrazena SPI komunikace, je zde připojeno jedno Master(řídící zařízení),což je například minipočítač a jedno Slave zařízení. Komunikace na sběrnici SPI probíhá takto:

- Master nastaví logickou 0 na SS zařízení, se kterým si chce vyměnit určité data
- Master začne generovat clk(hodinový signál) a hned poté vyšlou Master i Slave zařízení svá data
- Pro ukončení přenosu Master přestane vysílat clk a nastaví SS na logickou 1

#### UART:

Jedná se o zařízení pro asynchronní sériovou komunikaci. UART používá ke komunikaci 2 piny, jeden pro příjem dat (RX) a druhý pro odeslání dat (TX). Vysílání se začíná vysláním tzv. START bitu(po dobu jednoho bitu je signál změněn na logickou 0), poté se vyšlou potřebná data a ty jsou následovány STOP bitem(logickou jedničkou). Na obrázku 1.7 je zobrazen průběh komunikace v podobě odeslání jednoho B (Bytu),kde za STOP bitem může začít přenos dalšího Bytu. Sériová sběrnice na Raspberry Pi GPIO portu je přístupná na portu 14 a 15(Tx pro vysílání a Rx pro příjem dat). [7]



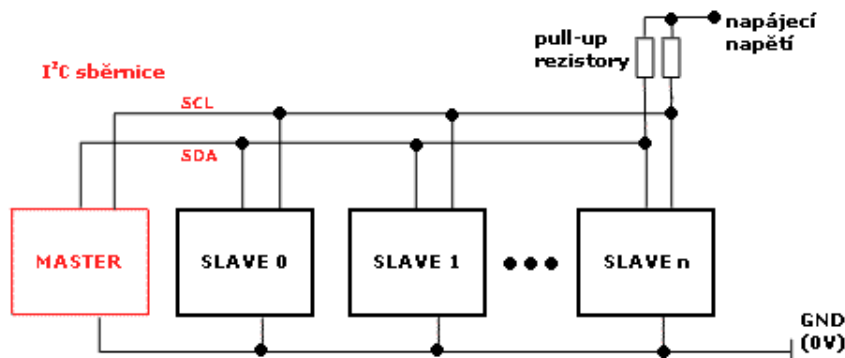
Obrázek 1.7: *UART přenos dat*

#### I<sup>2</sup>C sběrnice:

Tato sběrnice je určena pro komunikaci mezi více integrovanými obvody. Jedná se především o mikrokontroléry, sériové paměti, inteligentní LCD, audio a video obvody, a/d a d/a převodníky a některé další digitálně řízené obvody. Hlavní výhodou je, že obousměrný přenos probíhá pouze po dvou vodičích - "data SDA (serial data)" a "hodiny SCL (serial clock)". To především u mikrokontrolérů výrazně optimalizuje nároky na počet vstupně-výstupních pinů a celkově zjednodušuje výsledné zapojení. Na jednu sběrnici může být připojeno více integrovaných obvodů. Přenosová rychlost sběrnice je pro většinu aplikací dostatečná i v základní verzi, kde je frekvence hodin 100kHz. Ve vylepšených verzích to může být 400kHz nebo 1MHz.[5] [7]

Princip přenosu je takový,že jeden z integrovaných obvodů (většinou mikrokontrolér) je nastaven jako MASTER a všechny ostatní obvody jsou SLAVE. Obvody se dají zapojit i jako

tzv. multi-master, kdy je čipů master několik. Master při jakémkoli přenosu generuje hodinový signál na vodiči SCL. Když jeden čip vysílá, přijímají všechny ostatní a pouze podle adresy určí, zda jsou data určena jim. Čip, který chce vyslat/přijmout data musí nejprve definovat adresu čipu, s kterým chce komunikovat a zda půjde o příjem nebo vysílání - tedy o čtení nebo zápis. To určuje R/W (read/write) bit, který je součástí adresy. Na obrázku 1.8 můžeme vidět příklad zapojení pro komunikaci pomocí I<sup>2</sup>C sběrnice. [5]



Obrázek 1.8: I<sup>2</sup>C komunikace

## 1.4 Měřicí výbava

### 1.4.1 Payload box

Box, ve kterém je uložena všechna elektronika, včetně potřebných čidel, která je připojena k balónu se nazývá Payload box. Co vše bude obsahovat, záleží jen na návrhu projektu, co chceme, nebo co je pro nás důležité při měření získat. Popíšu zde základní věci, které se nejčastěji používají právě pro projekty balónů vypouštěných do velkých výšek. V měřicím boxu se nachází hlavní řídicí mikroprocesor, či minipočítač, ten řídí přenos dat a komunikuje s ostatními připojenými komponentami jako jsou GPS přijímač, rádiový modul, kamera a potom různá měřicí čidla, například teploměr, tlakoměr apod. A také je potřebné spolehlivé napájení v podobě baterií, které zajistí napájení celého boxu po dobu letu i dopadu, aby bylo možné poté nalézt balón s měřicím boxem pomocí neustálého zasílání GPS souřadnic.

### 1.4.2 Komponenty

Zde popisují jaké moduly, čidla a komponenty se používají v existujících projektech pro měření ve velkých výškách při návrhu měřicího boxu.

GPS modul:

Základní věc u měřicího boxu balónu, důležitá hlavně pro to pokud chceme měřicí box po přistání zpět na zem najít, jelikož budou právě v paměti řídicího počítače uloženy cenné naměřené data a především obrázky pořízené ve velké výšce. To nám umožní GPS přijímač, který získává ze satelitů údaje o aktuální poloze a ty poté předá počítači, který je dále zpracuje a ve většině případů přenáší data dále přes rádiové spojení na pozemní stanici a my podle souřadnic můžeme zjistit kde se box nachází. A nakonec nám informace o poloze měřicího balónu pomůže díky světovým souřadnicím najít na mapě místo dopadu a vystopovat tak pozici celého boxu s padákem po přistání zpět na zemi

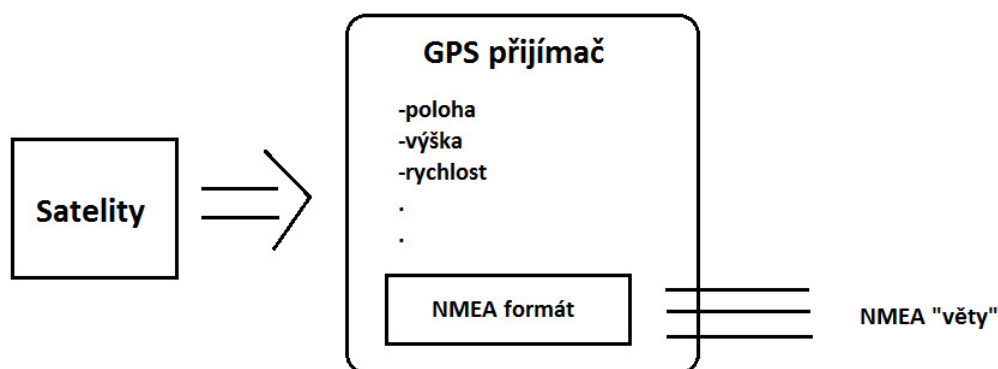
Kamera:

Další důležitou komponentou pro měřicí výbavu balónu je fotoaparát, nebo kamera, která by měla pořídít snímky z velké výšky, ty se potom uloží do paměti řídicího počítače a také se většinou posílají přes rádiové spojení na zem. Počítač bývá naprogramován aby pořídil a uložil snímky nebo videa ve vybraných časových intervalech.

### 1.4.3 GPS

Gps (Global positioning system) [6] je systém, který umožňuje kdekoliv na planetě Zemi určit přesnou zeměpisnou polohu pomocí speciálních družic, které obíhají planetu Zemi na oběžné dráze a vysílají elektromagnetické vlny. Jednotlivé družice jsou nastaveny tak, že všechny vyšlou signál v přesně danou dobu. GPS přijímač umístěný na Zemi (v našem případě v měřicím boxu pod balónem) vypočítá svou přesnou polohu podle toho s jakým zpožděním mu dojde signál z jednotlivých družic.

Většina GPS přijímačů pracuje podle norem NMEA [13], což je speciální formát dat, poté co GPS přijímač zjistí svou přesnou zeměpisnou polohu, pomocí satelitů převede zjištěné informace právě do tvaru formátu NMEA. Tyto data z GPS přijímače obsahují mnoho informací jako zeměpisnou šířku a délku, nadmořskou výšku, rychlost atd. Z GPS přijímače jsou potom odesílány data v podobě vět NMEA, celý přenos je znázorněn na obr. 1.9.



Obrázek 1.9: *funkce GPS přijímače*

Dekódování NMEA vět:

Protokol, kterým komunikuje přijímač, má přenosovou rychlost (baud rate) 4800, počet datových bitů je 8, přičemž sedmý bit (MSB) je vždy nulový. Počet stop bitů je jeden nebo více, parita není žádná. Navzájem spolu komunikuje vždy jeden mluvčí (talker) a jeden nebo více posluchačů (listeners). Veškerá data jsou posílána ve formě vět (sentences). Jsou dovoleny pouze tisknutelné ASCII znaky plus znaky konce řádku, tedy <CR> a <LF> (0x0d, 0x0a hexadecimálně). Každá věta začíná znakem \$ (dolar) a končí sekvencí <CR><LF>. Příkladem dotazovací věty může být následující sekvence: [13]

\$CCGPQ,GGA<CR><LF>

V této větě písmena CC označují počítač, který žádá přístroj GP (tedy GPS přijímač), aby zasílal věty typu GGA. Po této dotazovací větě by měl GPS přijímač zasílat každou sekundu větu typu GGA, dokud nedostane povel k zasílání jiného typu věty. Dvoupísmenných identifikátorů existuje mnoho, pro nás je však v této chvíli nejdůležitější, že pro GPS přijímače se používá identifikátor GP. Existuje nepřeberné množství různých vět, avšak většina gps přijímačů používá následující čtyři: [13]

- GSA,aktivní satelity

např. \$GPGSA,A,3,29,26,22,09,07,05,04,,,,,1.7,1.0,1.4\*30

-RMC,minimální doporučená informace pro navigaci

-GSV,informace o družicích

-GGA,zeměpisná délka a šířka,geodetická výška,čas určení souřadnic

## 2 Požadavky pro návrh řešení

Komplexním požadavkem pro návrh měřicí výbavy je modulárnost, možnost použití různých typů senzorů a využití jako cílové platformy minipočítač Raspberry Pi, který nabízí dostatečně možnosti propojení s externími moduly.

### 2.1 Použití Raspberry Pi

V mé práci se zabývám návrhem měřicího vybavení pro měření ve velkých výškách pomocí balónu s použitím tohoto malého počítače, který by měl řídit veškerou komunikaci. Ale proč vlastně použít právě Raspberry Pi?

Tento minipočítač má velmi kompaktní rozměry cca 85 x 56mm a nízkou hmotnost 45g. Tím pádem je vhodný pro použití v měřicím boxu, kde nebude zabírat příliš mnoho místa a váha celého boxu bude minimální. Další kladná vlastnost je malá spotřeba tohoto minipočítače, která činí dle oficiálních údajů u modelu B pouhých cca 2,5W, což je vzhledem k jeho výkonu a možnosti rozšíření velmi dobrá hodnota.

Jako základ, který bude organizovat komunikaci mezi jednotlivými moduly bude použit minipočítač Raspberry Pi ve variantě B, která nabízí větší operační paměť při zachování malé spotřeby energie a obsahuje navíc síťový konektor RJ-45, což zjednoduší komunikaci a ladění programů při připojení Raspberry Pi k počítači. Nyní budu porovnávat tento počítač s dalšími dostupnými jednodeskovými počítači, či mikrokontroléry, které se používají v podobných projektech.

Na Raspberry pi běží operační systém Linux(Raspbian), obsahuje 2 USB porty, hdmi výstup, můžeme k němu připojit monitor, nebo televizi a můžeme tedy na něm spustit jakékoliv programy, které běží pod systémem Linux a máme k dispozici uživatelské rozhraní z velkou softwarovou podporou, narozdíl od mikrokontrolérů bez operačního systému. Raspberry pi také nabízí připojení k síti a internetu, což je výhoda pro přenos dat a komunikaci např. s jiným klasickým počítačem. Také s ním můžeme pracovat s klávesnicí a myší, což je pro uživatele velká výhoda. Raspberry Pi používá jako datové úložiště paměťovou SD kartu, to je také výhodou, můžeme rychle změnit operační systém jen vyměněním karty. Za nízkou cenu je to velmi vhodné řešení k tomuto projektu, nabízí dostatečný výkon a jednoduchou obsluhu. Raspberry pi nabízí také možnost připojení kamery a to například přes USB, nebo speciální CSI kamerové sériové rozhraní.

### Raspberry vs Arduino:

Arduino[14] je malá programovatelná deska, obsahující mikrokontrolér ATmega 328, který dokáže přijímat a ukládat kódy z počítače. Je vhodný jen pro programování jednoduchých hardwarových projektů, na Arduino nelze nainstalovat operační systém, nemá tedy žádné uživatelské rozhraní ani video výstup. Další nevýhodou oproti Raspberry Pi je jeho výkon, procesor Arduina má frekvenci pouze 16Mhz oproti 700Mhz Raspberry Pi. Arduino neobsahuje klasické USB porty a nelze k němu připojit kameru, což je u tohoto projektu značná nevýhoda. Cena arduina je cca shodná s Raspberry Pi a proto je pro můj návrh výhodnější použít Raspberry pi, díky větším softwarovým možnostem a snadnějšímu uživatelskému přístupu.

### Raspberry vs BeagleBone Black:

Beagle Bone[15] je taková kombinace Raspberry Pi a Arduina, lze na něj nainstalovat operační systém linux. Nabízí také o něco výkonnější procesor 720Mhz, což ale na výsledném projektu nebude mít žádný vliv na funkčnost a Raspberry Pi svým výkonem postačuje. Cena BeagleBone je však oproti Raspberry pi více jak dvojnásobná, což není pro tento návrh výhodné, jelikož by ve výsledku neznamenal nárůst výkonu téměř žádnou výhodu. Cena Raspberry je cca 25 \$ oproti 89 \$ za BeagleBone.

## 2.2 Měřicí vybava pro Raspberry Pi

Raspberry Pi bude sloužit jako základní řídicí počítač, který bude komunikovat s připojenými moduly, s GPS přijímačem, kamerou a měřicími senzory. Musíme však uvážit použití nejvhodnějších komponentů právě ve spojení s tímto minipočítačem a aby bylo možné tyto moduly využívat v dosti odlišném prostřední stratosféry ve velké výšce.

Je potřeba použít GPS modul, který bude schopen pracovat ve velkých výškách nad 18 km. Pro použití ve spolupráci s minipočítačem Raspberry Pi bude použito připojení přes sériovou linku, která se nachází na GPIO konektoru na minipočítači. Toto řešení je vhodné zejména proto, že většina dostupných GPS modulů používá ke komunikaci právě sériový přenos informací. Dalším kritériem pro GPS přijímač je jeho pracovní teplota, jelikož při vypuštění do stratosféry bude muset snášet teploty i kolem -40 °C. Přijímač by měl mít také co nejmenší váhu a rozměry vhodné pro umístění do měřicího boxu a jeho spotřeba by také měla být co nejmenší, vzhledem k tomu, že celé zařízení bude napájenou pouze pomocí baterií s omezenou kapacitou.

Ohledně kamery(fotoaparátu) existuje více možností jaké použít. Pro připojení kamery k Raspberry Pi, můžeme využít připojení přes USB konektor, což nabízí mnoho komerčních webkamer, ale mnohem sofistikovanější řešení nabízí již v základu sám minipočítač, nabízí totiž speciální sériové rozhraní CSI(camera serial interface) ke kterému je možno připojit kameru, která byla vytvořena přesně k použití s počítačem Raspberry Pi. Tato možnost se velmi hodí do



mého návrhu řešení, jelikož je nejjednodušší a zároveň je vhodné pro využití kamery pro měření ve velkých výškách z důvodu spolehlivosti připojení přes CSI konektor..V CSI konektoru je totiž kamera připojena speciálním plochým kabelem, který se napevno uzavře do konektoru CSI umístěného na desce počítače Raspberry Pi.

Poslední hlavní částí části měřicí výbavy jsou čidla pro měření různých veličin, rozhodl jsem se použít teplotní čidlo pro měření okolní teploty. Při výběru čidla musím také brát v úvahu aby bylo schopno pracovat v nízkých teplotách a mohlo jednoduše komunikovat s počítačem Raspberry Pi.

Celá elektronika včetně řídicího počítače a přídatných modulů musí být napájena pomocí baterií, které dokážou udržet stabilní výkon i při nižší teplotě a mají dostatečnou kapacitu pro zásobování energií po dobu celého letu i následného udržení po dopadu boxu.

Dalším požadavkem je sestavení výsledného boxu, ve kterém bude umístěna měřicí výbava, ten by měl být zhotoven dostatečně pevný, aby ochránil elektroniku v případě nárazu a také by měl být dostatečně utěsněn, kvůli udržení vnitřní teploty v boxu, které bude dosaženo zahříváním řídicího počítače a ostatních komponent.

## 3 Návrh konkrétního řešení

### 3.1 Programovací jazyk

Pro komunikaci počítače Raspberry Pi a jednotlivých vybraných modulů je třeba naprogramovat řídicí aplikaci. Kód který bude napsaný v programovacím jazyce C se potom převede na strojový kód skládající se z nul a jedniček a ten půjde spustit z příkazového řádku na minipočítači. Pro kompilaci programů napsaných v jazyce C používám v Linuxu kompilátor nazvaný gcc, ten je již nainstalován na Raspberry Pi. Pro sestavení programu musím kompilátoru zadat zdrojové soubory a název spustitelného programu, který se vytvoří. Aby bylo možno použít kompilátor, je třeba přejít do adresáře, kde se nachází zdrojový soubor. Po zkompilování spustíme výsledný soubor zavoláním příkazu ./název\_souboru. Při práci s projektem obsahujícím více zdrojových souborů lze zjednodušit překlad pomocí nástroje make, který je také již nainstalován na Raspberry Pi, tento program hledá v adresáři soubor makefile a použije ho k provedení gcc příkazů.

#### 3.1.1 C pro Raspberry Pi

Pro tvorbu řídicí aplikace pro návrh měřicí výbavy pro Raspberry Pi, který bude komunikovat s ostatními použitými komponentami jsem vybral programovací jazyk C.

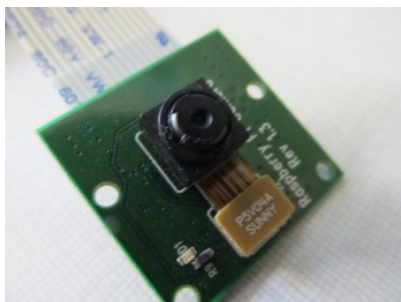
Jazyk C[8] je portabilní, v jazyce C je napsán i samotný Linux, který používá počítač Raspberry Pi jako operační systém(jeho upravenou verzi). Tento jazyk je velmi silný a použitelný téměř na všechny hardwarové platformy včetně Raspberry Pi. Další výhodou použití tohoto programovacího jazyka je dostupnost volných knihoven, které lze použít právě pro návrh programů pro komunikaci Raspberry Pi s jednotlivými moduly. Jazyk C jsem vybral také proto, že k němu mám větší osobní sympatie.

### 3.2 Použité komponenty

Pro vlastní návrh měřicí výbavy s ohledem na možnosti platformy počítače Raspberry Pi jsem vybral tyto moduly: Pro pořízení obrazu použiji kameru vytvořenou speciálně pro Raspberry Pi, jedná se o kameru Raspberry Pi Camera board. Dále pro sledování přesné polohy měřicího boxu a celého balonu použiji GPS přijímač RoyalTek, který je vhodný pro použití ve velkých výškách a pro měření teploty spolehlivý teploměr Dallas DS18B20.

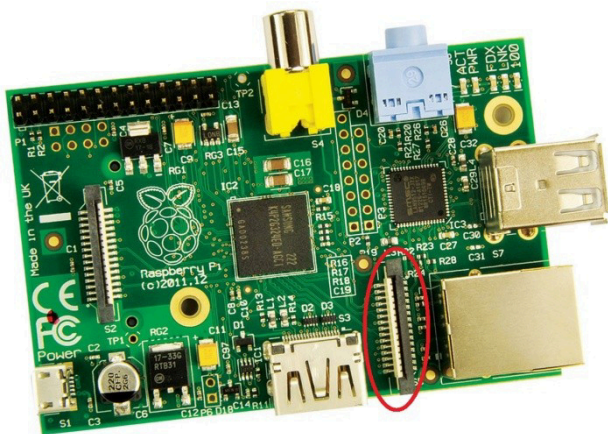
#### 3.2.1 Raspberry Pi Camera Board

Raspberry Pi Camera board byla vyvinuta speciálně pro počítač Raspberry Pi firmou Raspberry Pi Foundation, má velmi malé rozměry a zanedbatelnou hmotnost při zachování velmi dobrých funkčních parametrů, tím pádem je vhodná pro použití v měřicím boxu.



*Obrázek 3.1: Raspberry Pi Camera board*

Raspberry Pi Camera Board[16] nabízí rozlišení 5MP ( $2592 \times 1944$  pixelů) s použitím čidla Omnivision 5647. Modul se připojuje na Raspberry Pi s použitím 15 pinového plochého ribbon kabelu k vyhrazenému 15-pinovému sériovému rozhraní CSI(camera serial interface, obr 3.2), který byl navržen speciálně pro připojení kamer. Deska modulu kamery je velmi malá, má kolem 25 mm x 20 mm x 9 mm a váží jen něco málo přes 3g, takže je ideální pro mobilní nebo jiné aplikace, kde je důležitá velikost a hmotnost. Samotné čidlo má nativní rozlišení 5 megapixelů a má pevný objektiv upevněný na desce modulu. Z hlediska statických snímků, je fotoaparát schopen zaznamenat obrázek o velikosti 2592 x 1944 obrazových bodů, a také podporuje 1080p @ 30fps, 720p @ 60fps a 640x480p 60/90 nahrávání videa.



*Obrázek 3.2: Konektor CSI na Raspberry Pi*

Ve srovnání s kamerami připojených přes rozhraní USB tato kamera využívá při záznamu grafické jádro přímo na Raspberry Pi VideoCore a nezatěžuje tak samotný procesor počítače Raspberry Pi. Další velkou výhodou je možnost použití již nainstalovaných programů v operačním systému Raspbian pro záznam videa a snímků z kamery Raspberry Pi camera board.

Pro příklad uvádím srovnání s USB kamerou Logitech HD C270[17] s přibližně stejnou cenou, tato USB kamera má menší rozlišení videa 1280 x 720 px 30 sn./s oproti raspberry camera board ,která má až 1920×1080 –30fps a také menší rozlišení fotografií. Také je rozměrově větší a tím pádem méně praktická na upevnění do boxu. Ale hlavně při pořizování snímků nemůže využít grafického jádra na Raspberry Pi a vytěžováním CPU by mohlo dojít při současném běhu všech čidel k pádu programu a následné ztrátě spojení a naměřených dat.

#### Použití Raspberry Pi camera board:

Operační systém Raspbian nabízí po povolení kamery v nastavení příkazem raspiconfig možnost řídit tuto kameru pomocí 2 vytvořených programů a to raspistill pro pořízení statických snímků do formátu jpg a raspivid pro záznam videa ve formátu h264. U obou programů je možnost změnit různé hodnoty kamery a dobu po kterou bude natáčet video, či za jak dlouho má být pořízena fotografie.[16]

#### Raspistill:

Formát pro použití tohoto příkazu je: raspistill [volby], kde namísto volby můžeme přidat daný parametr, z těch nejpoužívanějších se jedná např o:

-co ,nastavení kontrastu fotografie(-100 až 100)

-w ,nastavení šířky fotografie

-h ,nastavení výšky

-q ,nastavení kvality jpeg(0-100)

-o ,nastavení výstupu, název výstupního souboru

-hf ,horizontální převrácení fotografie

př. pro pořízení fotografie s otočením slouží příkaz: raspistill -vf -o image.jpg [16]

#### Raspivid:

Formát pro použití tohoto příkazu je stejný jako u raspivid tedy: raspivid [volby], jako předvolené parametry lze použít:

-stejně jako u raspistill -w,-h,-o,-co,-hf,-wf

-rot ,rotace videa(90,180,360)

-fps ,specifikace počtu snímků za sekund

př.pro pořízení 30s videa ve VGA rozlišení a 30fps:

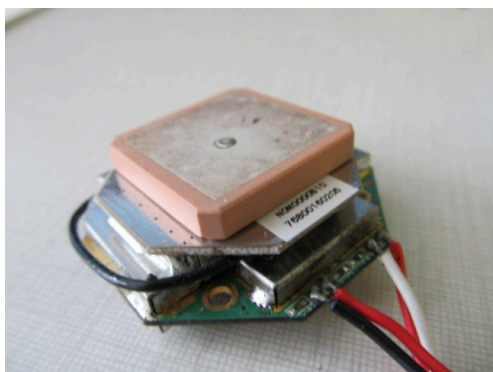
raspivid -w 640 -h 480 -fps 30 -t 30000 -o video.h264

seznam všech možných parametrů nám poskytne příkaz: raspivid | less [16]

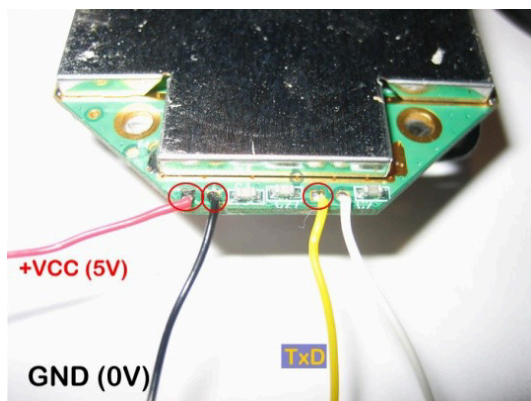
### 3.2.2 GPS modul RoyalTek RGM-2000

Pro určení aktuální geografické polohy měřicího boxu použijí GPS přijímaček Royaltek RGM - 2000[15] od společnosti Sapphire.

Tento integrovaný GPS přijímač dokáže zaměřit přesnou polohu na Zemi do 60 vteřin od zapnutí. Jeho výhodou jsou jeho velmi malé rozměry(50x62mm), nízká hmotnost cca 40g a proto je vhodný pro použití do měřicího boxu. Jeho pracovní teplota se může pohybovat mezi -40°C až 85°C a nebude problém použití tohoto modulu pro měření ve stratosféře za nízké teploty. Dále tento modul obsahuje aktivní anténu pro kvalitnější a rychlejší příjem signálu ze satelitů. S počítačem Raspberry Pi se propojí pomocí sériové linky, která je umístěna na rozšiřujícím GPIO konektoru. Pro napájení modulu je potřeba 5V, což nám zajistí opět výstup na pinu GPIO portu. Celý modul je zobrazen na obrázku 3.3 na obrázku 3.4 jde vidět zapojení vodičů napájení a vysílacího vodiče (TxD), který je napojen na příjem sériové linky na Raspberry Pi(RxD) [9]



*Obrázek 3.3: GPS přijímač RoyalTek*



*Obrázek 3.4: rozmístění pinů*

Tento GPS modul jsem vybral také proto, že jsem ho měl zrovna k dispozici a jeho parametry dostačují na měření ve velkých výškách i jeho rozměry a váha jsou ideální pro použití v měřicím boxu.

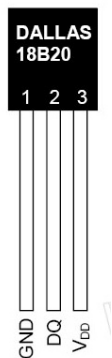
Tento GPS přijímač vypočte aktuální zeměpisnou polohu z dostupných satelitů a přes sériovou linku posílá NMEA věty, které se poté dekodují na Raspberry a získává se aktuální zeměpisná šířka, délka a výška.

### 3.2.3 Teploměr Dallas DS18B20

Toto zařízení je velmi malý digitální teploměr, pro použití v měřicím boxu jsem ho vybral právě díky jeho jednoduchosti připojení přes jeden datový vodič, miniaturním rozměrům a téměř žádné váze. Jak vypadá teploměr reálně můžete vidět na obr.3.5 a na obrázku 3.6 je schéma zapojení jednotlivých pinů.



*Obrázek 3.5: Teploměr DS18B20*



Obrázek 3.6: piny teploměru

Tento teploměr nabízí pro měření 9b -12b ve stupních Celsia.. Pro komunikaci používá tzv. sběrnici 1-Wire(v překladu jeden vodič), to znamená že pro přenos dat obsahující aktuální teplotu slouží pouze jeden datový vodič, což je výhoda i z hlediska použití počítače Raspberry Pi, jelikož k němu bude teploměr připojen na jeden vstupně výstupní GPIO pin. Další výborná vlastnost teploměru je že dokáže měřit teplotu v rozmezí  $-55^{\circ}\text{C}$  až  $125^{\circ}\text{C}$  a to je dostačující pro použití ve velké výšce ve stratosféře, kde se pohybuje okolní teplota okolo  $\pm 40^{\circ}\text{C}$ . Tento teploměr je velice rozšířený a operační systém Raspbian pro něj již nabízí ovladače pro načtení informace s teplotou.[18]

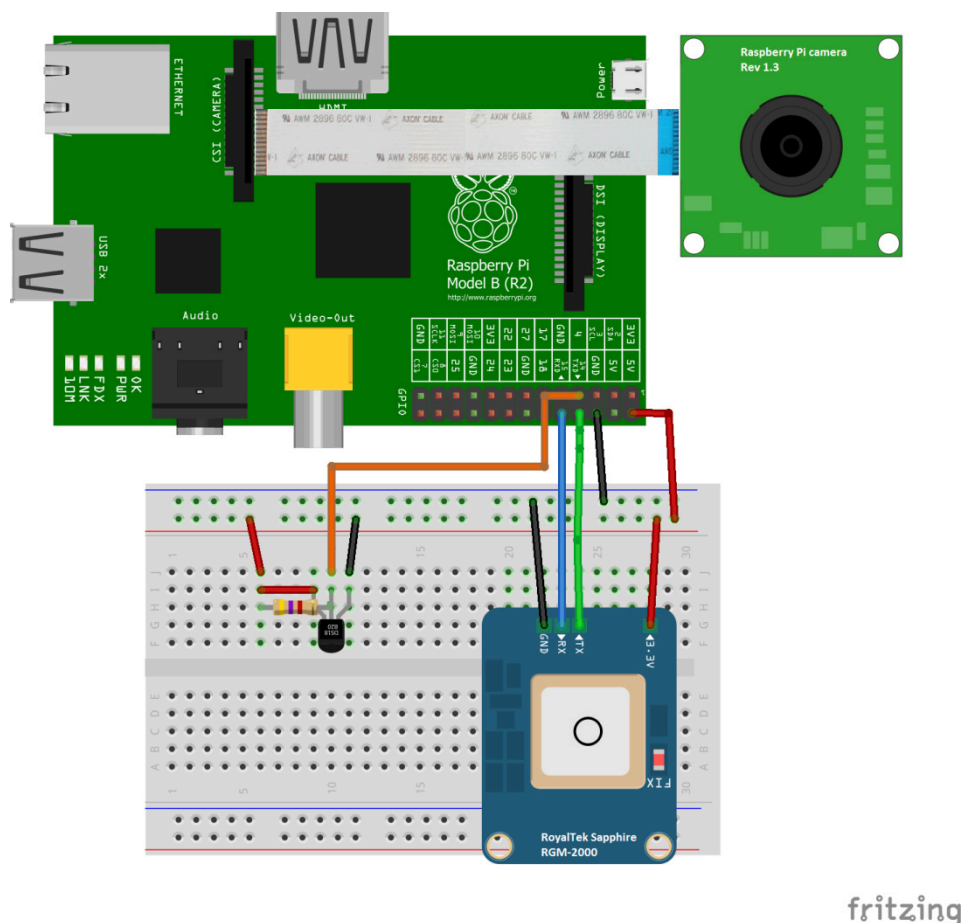
Teploměr má své unikátní 64bit číslo, které mu umožňuje na jedné sběrnici provozovat i více takových teploměrů zároveň, což žádné jiné teploměry nenabízí, v mém návrhu jsem se rozhodl ale použít jen jeden, jelikož je to dostačující pro změření okolní teploty ve vzduchu a není potřeba ji měřit na více místech. Teploměr vyžaduje napájecí napětí 3-5V, takže můžeme využít jak 3,3V tak 5V napájecí pin na GPIO portu Raspberry Pi.

Při zapojení teploměru je potřeba použít jeden 4.7K pullup rezistor, který se zapojí mezi napájecí vodič vcc a datový vodič dq. Tento odpor zajistí stabilní příjem dat na Raspberry Pi. Samotný chip teploměru obsahuje 1-wire rozhraní, stejně jako logiku ovládání a snímač. Operační systém Raspbian zvládá pomocí programu dekodovat zprávy s teplotou vysílané čidlem.

### 3.2.4 Celkové zapojení

Zde je znázorněno schéma celkového zapojení měřicího boxu s řídicím počítačem Raspberry Pi a k němu připojenými komponentami: GPS modulem, kamerou a teploměrem.

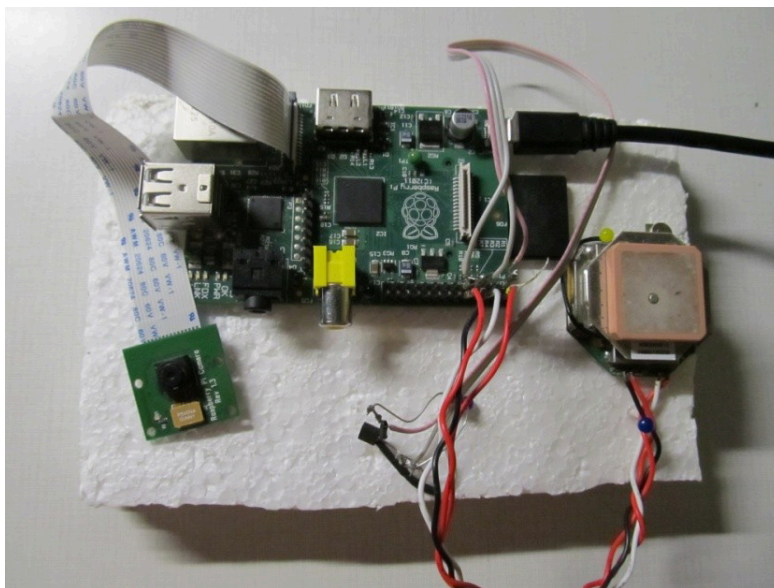
Kamera Raspberry Pi camera board je připojena přes speciální CSI sériový port přímo na desce Raspberry Pi. GPS modul je připojen na GPIO port na UART(sériovou linku) a teploměr je připojen pomocí sběrnice 1- Wire k jednomu pinu na portu GPIO. Zapojení je zobrazeno na obr.3.7, schéma bylo nakresleno v programu Fritzing[19] a upraveno.



Obrázek 3.7: Celkové zapojení

Reálné názorné zapojení počítače Raspberry Pi s externími komponenty jsem zdokumentoval a je vidět na obrázku 3.8, kde můžeme vidět samotný minipočítač Raspberry Pi model B s připojeným GPS modulem, kamerou Raspberry Pi camera board a teploměrem DS18B20. Počítač Raspberry Pi je napájen přes napájecí konektor micro usb napětím 5V.





*Obrázek 3.8: celkové zapojení*

### 3.3 Řídící aplikace

Aby počítač Raspberry Pi mohl komunikovat se všemi připojenými komponentami, potřebuje k tomu řídící aplikaci(počítačový program), je třeba naprogramovat komunikaci, já se zaměřím na programování aplikace pro řízení kamery, GPS modulu a teploměru.

Celá aplikace je rozdělena do jednotlivých modulů, z nichž každý má svůj zdrojový soubor, ve kterém jsou implementovány funkce pro komunikaci s daným modulem. Základním jádrem je program(zdrojový soubor main.c), do kterého jsou nainportovány jednotlivé soubory modulů(GPS přijímač, teplotní čidlo a kamera). V tomto jádře programu se volají metody přídatných modulů a tím se získají potřebné data, které se poté ukládají do jednoho souboru(logFile.txt). V souboru jsou odděleny vždy jednotlivé bloky naměřených hodnot obsahující gps souřadnice, výšku a teplotu ve stupních Celsia. Pro snadnější práci s naměřenými hodnotami ukládám ještě navíc zvlášť do souboru jen gps pozici(gpsFile.txt) a do hodnoty teploty(tempFile.txt). Zde je ukázka části programu z hlavní funkce main.c:

```

/// hlavni smyčka,defaultne nastaveny nekonecny cyklus
if(pom==0)
{
    while (1)
    {
        startfile(pom2)
        write_to_fileGPS();
        write_to_fileTEMP();

        if(pom2%p1 == 0)
        {
            photo(b,b1,pic);
            b1++;
        }
        sleep(s1

        if(videnable==1
        {
            printf("zaznam videa\n");
            system(vid1);
            videnable = 0;
        }
        pom2++;

    }
} ...

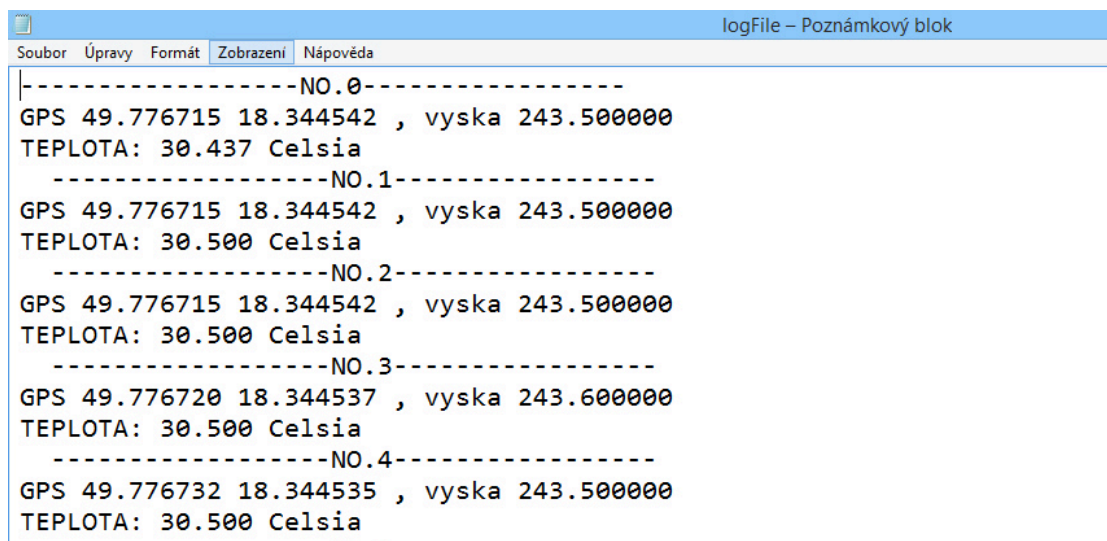
```

V této hlavní funkci běží smyčka ve while(), ta se na začátku programu nastaví buď na nekonečný cyklus (proměnná pom = 0), nebo konečný cyklus s daným počtem opakování (pom = počet cyklů). V každém cyklu této smyčky se volají funkce pro načtení dat ze senzorů (GPS přijímač a teploměr) a načtené hodnoty se ukládají do společného souboru logFile.

Podle nastavení je také v každém daném cyklu zaznamenána fotografie,nebo video(dle zvoleného nastavení). Funkce pro zápis do souboru je zobrazena na následujícím příkladu kódu:

```
void write_to_fileGPS()
{
    gps_location(&data);    // nacteni dat z gps
    pFile3 = fopen (logname,"a");
    fprintf (pFile3,"GPS %lf %lf , vyska %lf \n",
    data.latitude, data.longitude, data.altitude);
    printf("GPS %lf %lf , vyska %lf\n", data.latitude,
    data.longitude, data.altitude);
    fclose (pFile3);
    pFile = fopen (gpsname,"a");
    fprintf (pFile,"%lf %lf\n", data.latitude,...
    fclose (pFile);
}
```

Tato funkce načte data s GPS přijímače pomocí funkce `gps_location(&data)`, poté otevře textový soubor pro přidávání a запиše údaje jako jsou zeměpisná délka, šířka a výška,po zápisu je soubor uzavřen. Ukázka souboru `logFile.txt` je na obrázku 3.9



*obr.3.9: logFile.txt*

### 3.3.1 Pořízení obrazu

Pro pořízení statických snímků i videa pomocí kamery Raspberry Pi camera board používám pro tuto kameru již existující funkce raspistill a raspivid, které jsou již dostupné v operačním systému Raspbian. Pro volání těchto funkcí v příkazovém řádku Linuxu používám volání funkce system("prikaz ..."). Tato funkce spustí příkaz, který je uveden jako její parametr typu řetězec a čeká dokud příkaz nebude vykonán. Zde je ukázka pro záznam fotografií v jpg formátu:

Nejprve určím deklaraci typu řetězec, která obsahuje celý název příkazu, který se má zavolat na operačním systému Linux.

```
char pic[] = "raspistill -o pict1.jpg -t 500";
```

Toto je deklarace příkazu, který provede záznam fotografie s názvem pict1 ve formátu jpg a záznam se provede po časové prodlevě 500ms. Na další ukázce je funkce, která provede vykonání tohoto příkazu.

```
void photo(char *b, int m, char *p2)
{
    sprintf(b, "%d", m);

    char *s = replace(p2, '1', b);
    printf("photo pict%d\n", m);
    system(s);
    free(s);
}
```

V této funkci se příkazem system(s) provede příkaz uvedený v řetězci s, který je předán v parametru funkce. Dále se zde provádí pomocí metody replace(...) v daném řetězci vždy změna v názvu fotografie, aby se při dalším volání funkce photo nepřepsala jinou fotografií se stejným názvem.

### 3.3.2 Získání GPS souřadnic

Je potřeba vytvořit také funkce které získají za pomoci připojeného GPS přijímače souřadnice aktuální polohy. Program musí zpracovat přijímané NMEA věty a z nich potom lze vyčíst aktuální zeměpisnou šířku, délku a výšku..

Pro dekódování NMEA vět, které obsahují GPS souřadnice používám funkce s již existující knihovny[10].

GPS přijímač posílá data na Raspberry Pi pomocí sériové sběrnice UART, ta se pod operačním systémem na Raspberry Pi objeví jako soubor /dev/ttyAMA0. Pro použití UART na Raspberry Pi se musí upravit soubor /boot/cmdline.txt , úpravu tohoto souboru lze provést použitím již připraveného skriptu , použití je popsáno v příloze na CD v souboru readme.txt. Nyní popíšu vybrané funkce pro záznam s GPS modulu, které se nacházejí ve zdrojovém souboru gps.c.

```
void serial_config(void)
{
    struct termios options;
    tcgetattr(uart0_filestream, &options);
    options.c_cflag = B4800 | CS8 | CLOCAL | CREAD;
    options.c_iflag = IGNPAR;
    options.c_oflag = 0;
    options.c_lflag = 0;
    tcflush(uart0_filestream, TCIFLUSH);
    tcsetattr(uart0_filestream, TCSANOW, &options);
}
```

Tato funkce konfiguruje sériové rozhraní UART, pro příjem dat s GPS je nejdůležitější nastavit Baud rate(rychlost sériové linky) na 4800 Baud, velikost přenosu dat po 8b(CS8), ignorování znaků s chybnou paritou(IGNPAR).

```
void nmea_parse_gpgga(char *nmea, gpgga_t *loc)
{
    char *p = nmea;
    p = strchr(p, ',')+1; //skip time
    p = strchr(p, ',')+1;
```

```

loc->latitude = atof(p);

p = strchr(p, ',')+1;
switch (p[0]) {
    case 'N':
        loc->lat = 'N';
        break;
    case 'S':
        loc->lat = 'S';
        break;
    case ',':
        loc->lat = '\0';
        break;
}

```

Zde je ukázka funkce, která dekoduje světové souřadnice z přijatých NMEA vět z GPS přijímače. Parsuje se tady věta GGA, ze které se získá zeměpisná šířka, délka a výška tyto údaje jsou uloženy do datové struktury loc.

### 3.3.3 Čtení teploty

Pro měření teploty teplotním čidlem DS18B20 jsem napsal program, který čte aktuální teplotu. Při návrhu funkcí mi pomohl příspěvek Matta Hawkinse viz odkaz [11].

Tento kód je založen na dvou modulech jádra, které musí být načteny před spuštěním následujícího kódu pomocí příkazů:

```

sudo modprobe w1-gpio
sudo modprobe w1-therm

```

Tyto příkazy je výhodné umístit do spouštěcího skriptu, aby se načetly hned po spuštění operačního systému na Raspberry Pi.

Teploměr je připojen k Raspberry Pi pomocí 1-wire sběrnice a po načtení ovladačů se nachází zařízení v adresáři: sys/bus/w1/devices, teploměr má své unikátní ID číslo ve tvaru

28 - číslo. V tomto adresáři se nachází složka w1\_slave, která obsahuje aktuálně naměřená data a záznam t = xx udává teplotu ve stupních Celsia. Přikládám funkci pro načtení teploty do proměnné tempC umístěnou ve zdrojovém souboru temperature.c.

```
float readtemp(char *devPath, char *buf, char *tmpData)
{
    float tempC;
    ssize_t numRead;
    int fd = open(devPath, O_RDONLY);
    if(fd == -1)
    {
        perror ("Couldn't open the w1 device.");
        return 1;
    }
    while((numRead = read(fd, buf, 256)) > 0)
    {
        strncpy(tmpData, strstr(buf, "t=") + 2, 5);
        tempC = strtod(tmpData, NULL);
        close(fd);

        return tempC;
    }
}
```

Funkce čte teplotu z adresáře se zařízení teploměru a ukládá jí, později je teplota pomocí funkce writetofileTEMP uložena do společného souboru logFile.txt.

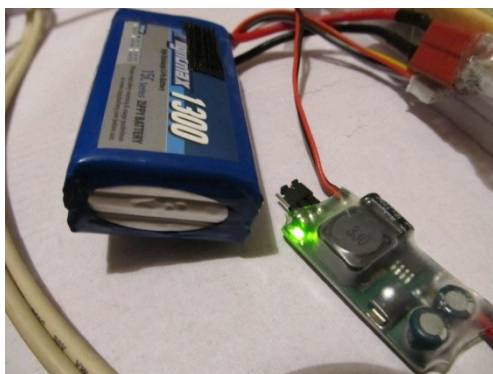
## 4 Testování

V této poslední kapitole popisují provedené testy měřicího boxu v simulovaných podmínkách, kvůli ověření funkčnosti všech jednotlivých modulů. Dále zde uvedu informace o použitém napájení boxu a naměřené údaje o spotřebě jednotlivých komponent i kompletního zapojení.

### 4.1 Napájení

K napájení minipočítače Raspberry Pi je zapotřebí stabilní napětí 5V a dostatečný povolený odběr proudu minimálně cca 1A.

Pro můj návrh využívám pro testovací účely jako napájení Li-Po baterii Lightmax 1300, která má výstupní napětí 11V a maximální povolený odběr proudu až 20A, což je s velkou rezervou dostačující. Tuto baterii jsem použil, jelikož jsem ji měl zrovna k dispozici a vyhovovala požadavkům této práce, baterie má také malé rozměry cca 7cm x 3cm x 1cm a váhu cca 100g., k baterii používám konvertor napětí na 5V, na výstupu konvertoru je umístěn usb port, který je poté propojen s napájecím micro usb portem na Raspberry Pi.. Baterie s konvertorem je zobrazena na obrázku 4.1.



*obr.4.1: baterie a konvertor*

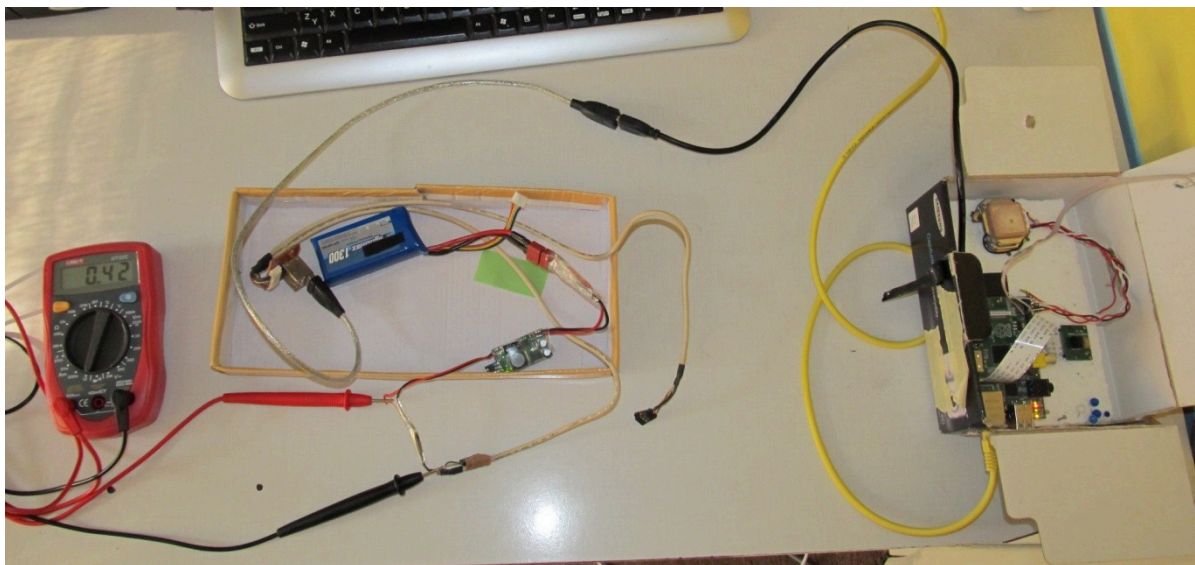
Pomocí zapojení s měřicím přístrojem UNI-T UT3C jsem provedl orientační měření odebíraného proudu při spuštění jednotlivých komponent. Zde uvádím výsledky měření (uvedeno co bylo zapojeno a změřený odběr elektrického proudu v mA).



Raspberry Pi(samotný)	- 390mA
RaspberryPi + camera board(záznam fotografie)	- 530mA
RaspberryPi + camera board(záznam videa)	- 660mA
RaspberryPi + GPS modul	- 470mA
RaspberryPi + camera board(záznam fotografie) + GPS modul	- 620mA
RaspberryPi + camera board(záznam videa) + GPS modul	- 740mA
Program(GPS modul,zapínání kamery v intervalech)	-520mA

Teploměr zde neuvádím,jelikož jeho připojením nezaznamenal měřicí přístroj žádný zvýšený odběr proudu a jeho spotřeba je tedy zanedbatelná.

Při běžném běhu programu s pořizováním fotografií po určité době a neustálého běhu GPS modulu a teploměru dosahoval maximální proud hodnoty cca 520mA, což při použití dané baterie s kapacitou 1300mA dává provozní výdrž na baterii cca 2h 30 minut. Testovací měřicí zapojení je zobrazeno na fotografii 4.2. Měřicí přístroj je zapojen sériově mezi kladnou svorku baterie a konvertor napětí.



*obr.4.2 měření odběru proudu*

## 4.2 Testování v simulovaných podmínkách

Na závěr jsem provedl pár testů jako ukázkou funkčnosti celého zapojení a jednotlivých komponent.

Nejprve jsem provedl test funkčnosti za nižší teploty, Raspberry Pi jsem umístil spolu s ostatními moduly do pevné krabičky a umístil ji do mrazáku na 45 min, teplota klesla pod -7 stupňů Celsia a všechny komponenty byly plně funkční. Uvádím zde fotografii měřicího zařízení umístěného v mrazáku (obr.4.3) a ukázkou ze souboru (obr.4.4), ve kterém se ukládají naměřená data. Celý soubor je poté přiložen v příloze na CD.



obr.4.3: měřicí box v mrazáku

```
-----NO.267-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.000 Celsia
-----NO.268-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.062 Celsia
-----NO.269-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.062 Celsia
-----NO.270-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.062 Celsia
-----NO.271-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.062 Celsia
-----NO.272-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.062 Celsia
-----NO.273-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.062 Celsia
-----NO.274-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.125 Celsia
-----NO.275-----
GPS 0.000000 0.000000 , vyska 0.000000
TEPLOTA: -6.125 Celsia
```

obr.4.4 soubor logFile.txt

V dalším testu jsem měřil všechny údaje a pořizoval fotografie, provedl jsem v automobilu krátkou testovací jízdu a krabičku s měřicí výbavou jsem měl nainstalovanou za čelním sklem na palubní desce, po dobu jízdy cca 10 min v nízké rychlosti jsem zaznamenával GPS souřadnice, teplotu ve voze, kterou jsem v rámci testu pomocí topení reguloval a tím vystavil měřicí vybavení i vyšším teplotám nad 40 stupňů Celsia. Po dobu jízdy se také ve stanovených intervalech zaznamenávali fotografie z kamery. Test proběh bez problémů, akorát GPS navigaci trvalo delší dobu (cca 3 minuty), než zachytila ze satelitů dostatečný signál. Uvádím ukázkovou fotografii pořízenou kamerou během jízdy, fotografie jsou pořízeny v plném rozlišení v jpg formátu. (obr.4.5). Ostatní fotografie z jízdy jsou umístěny v příloze na CD.



*obr.4.5 fotografie během jízdy*

Dále přikládám ukázkou souboru s naměřenými daty z testovací jízdy (obr.4.6), celý soubor je také součástí přílohy na CD.

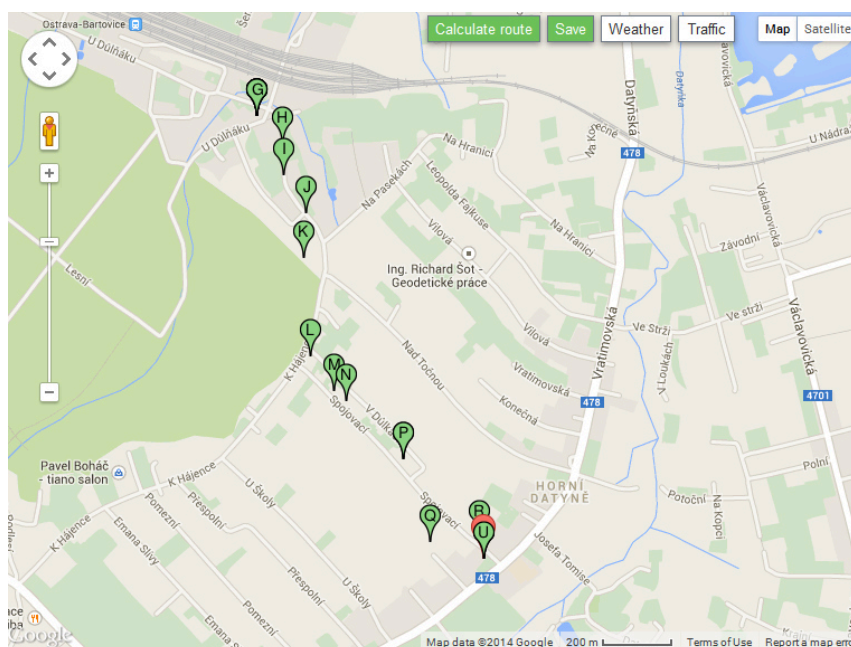
```

GPS 49.776820 18.345280 , vyska 247.600000
TEPLOTA: 40.937 Celsia
-----NO.168-----
GPS 49.776813 18.345322 , vyska 249.200000
TEPLOTA: 40.937 Celsia
-----NO.169-----
GPS 49.776805 18.345342 , vyska 250.200000
TEPLOTA: 41.000 Celsia
-----NO.170-----
GPS 49.776793 18.345385 , vyska 249.600000
TEPLOTA: 41.062 Celsia
-----NO.171-----
GPS 49.776775 18.345432 , vyska 248.100000
TEPLOTA: 41.062 Celsia
-----NO.172-----
GPS 49.776750 18.345493 , vyska 247.700000
TEPLOTA: 41.125 Celsia
-----NO.173-----
GPS 49.776722 18.345557 , vyska 248.600000
TEPLOTA: 41.125 Celsia
-----NO.174-----
GPS 49.776690 18.345633 , vyska 249.300000
TEPLOTA: 41.125 Celsia
-----NO.175-----
GPS 49.776658 18.345705 , vyska 250.500000
TEPLOTA: 41.125 Celsia
-----NO.176-----

```

obr.4.6 ukázka souboru logFile.txt

Ze získaných GPS souřadnic jsem na mapě(www.googlemaps.com) znázornil jednotlivé naměřené body(obr.4.7), které jsou téměř všechny velmi blízko silnice, po které jsem jel testovacím autem a přesnost GPS přijímače je tedy velmi dobrá.



obr.4.7 trasa



## Závěr

Cílem této bakalářské práce bylo navrhnout komplexní měřicí výbavu pro balón, který by byl schopen letět do velké výšky a získat potřebná data, zároveň při tomto návrhu bylo stanoveno použití platformy Raspberry Pi.

V této práci byla probrána tematika měření ve velkých výškách pomocí balónu. V teoretické části byla popsána problematika vypouštění měřících balónů do stratosféry, podrobně bylo popsáno z čeho se skládá měřicí sonda, kterou vynáší balón do velké výšky. Dále byl popsán minipočítač Raspberry Pi, který slouží jako základ pro návrh měřicí výbavy v této práci, jeho možnosti použití a jaké komponenty a pomocí kterých rozhraní se k tomuto počítači dají připojit. Jaké operační systémy jsou k dispozici pro Raspberry Pi.

Po teoretické části následuje popis požadavků na měřicí výbavu s ohledem na možnosti platformy počítače Raspberry Pi. Jaké druhy komponent jsou nejvhodnější pro získávání informací v náročnějším prostředí ve stratosféře a zároveň nevhodnější pro spolupráci s minipočítačem Raspberry Pi. Proč byl vybrán právě jednodeskový počítač Raspberry Pi, zároveň je tento minipočítač porovnán s konkurenčními podobně zaměřenými jednodeskovými počítači, či mikrokontroléry.

V další části jsem navrhoval použití konkrétních existujících komponent pro měření ve velkých výškách, navrhnul jsem testovací zapojení GPS přijímače pro získávání aktuální polohy, kamery pro vytváření snímků a teplotního senzoru na měření okolní teploty. Tento výběr komponent jsem zdůvodnil a porovnal s konkurencí. Dále jsem napsal řídicí aplikaci, která umožňuje jejich vzájemnou komunikaci s počítačem Raspberry Pi a ukládání naměřených hodnot z jednotlivých čidel a snímků z kamery.

V poslední části jsem provedl testování navrženého zapojení, všechny komponenty jsou plně funkční. Zdokumentoval jsem testy v simulovaných podmínkách. Měřil jsem za nízké i vysoké teploty, pořídil jsem několik fotografií v terénu a naměřil jsem hodnoty zeměpisné polohy pomocí GPS na testovací trase autem. Zeměpisnou polohu testovací trasy jsem znázornil na mapě. Dále jsem změřil spotřebu elektrické energie měřícího modulu v různých režimech zapojení.

Další rozšíření této práce by mohlo být použití rozsáhlejší měřicí výbavy s dalšími přídatnými komponentami. Také řídicí aplikace by mohla obsahovat více funkcí. Sonda odesílat fotografie a data za letu zpět na zem v reálném čase pomocí rádiového signálu.. V budoucnu by bylo vhodné práci rozšířit a dovést k praktické realizaci vypuštění balónu s měřícím boxem.

## Použitá literatura

- [1] Layers of the Atmosphere. *National weather service* [online]. [cit. 2014-07-06]. Dostupné z: <http://www.srh.noaa.gov/srh/jetstream/atmos/layers.htm>
- [2] Isas Japan aerospace exploration agency. *Research on Balloons to float over 50km altitude* [online]. [cit. 2014-07-13]. Dostupné z: <http://www.isas.jaxa.jp/e/special/2003/yamagami/03.shtml>
- [3] Project Horus. *High altitude balloon* [online]. [cit. 2014-07-06]. Dostupné z: <http://projecthorus.org/>
- [4] Projekt Fénix. Technické provedení [online]. 19.3.2014 [cit. 2014-05-04]. Dostupné z: <http://www.fenix.svetakraj.cz/p/technicke-provedeni.html>
- [5] HIMPE, Vincent. *Mastering the I2C Bus*. Elektor International Media, 2011. ISBN 090570598X, 9780905705989.
- [6] *Global positioning system standart positioning service signal specification: 2 nd edition*. Navigation Center, 1995. Dostupné z: <http://www.navcen.uscg.gov/pubs/gps/sigspec/gpssps1.pdf>
- [7] EBEN, Upton; HALFACREE, Gareth. *Raspberry Pi User Guide*. Wiley, United Kingdom, 2012. ISBN 978-1-118-46446-5
- [8] KADLEC, Václav. *Učíme se programovat v jazyce C*. Brno, CP Books, 2005. ISBN 80-7226-715-9
- [9] Saphire GPS Receiver RoyalTek RGM-2000. Tao Yuan City, Taiwan, 2002.
- [10] Walter Dal Mut. *Raspberry Pi – C GPS NMEA library* [online]. 2013 [cit. 2014-05-05]. Dostupné z: <http://walterdalmut.com/2013/11/23/raspberry-pi-c-gps-nmea-library-global-positioning-system/>
- [11] Raspberry Pi Spy. *Raspberry Pi 1- Wire Digital Thermometer Sensor* [online]. 2013 [cit. 2014-05-06]. Dostupné z: <http://www.raspberrypi-spy.co.uk/2013/03/raspberry-pi-1-wire-digital-thermometer-sensor/>
- [12] . *BCM2835 ARM peripherals* [online]. 2012 [cit. 2014-07-13]. ISBN CB4 0WW. Dostupné z: <http://www.raspberrypi.org/wp-content/uploads/2012/02/BCM2835-ARM-Peripherals.pdf>
- [13] BETKE, Klaus. *The NMEA 0183 Protocol* [online]. 2001 [cit. 2014-07-13]. Dostupné z: [http://www.agt.bme.hu/tantargyak/bsc/bmeoafav49/NMEAdescription\\_gy\\_12.pdf](http://www.agt.bme.hu/tantargyak/bsc/bmeoafav49/NMEAdescription_gy_12.pdf)
- [14] DURFEE, W. *Arduino Microcontroller Guide* [online]. 2011 [cit. 2014-07-13]. Dostupné z: <http://www.me.umn.edu/courses/me2011/arduino/arduinoGuide.pdf>
- [15] COLEY, Gerald. *Beaglebone black system reference manual* [online]. 2013 [cit. 2014-07-13]. Dostupné z: [http://www.adafruit.com/datasheets/BBB\\_SRM.pdf](http://www.adafruit.com/datasheets/BBB_SRM.pdf)
- [16] FOUNDATION, Raspberry Pi. *Raspicam documentation* [online]. 2013 [cit. 2014-07-13]. Dostupné z: <http://www.raspberrypi.org/wp-content/uploads/2013/07/RaspiCam-Documentation.pdf>

- [17] *HD webcam C270* [online]. 2014 [cit. 2014-07-13]. Dostupné z: <http://www.galaxus.ch/Files/5/9/4/6/1/196727.pdf>
- [18] *DS18B20 Digital Thermometer* [online]. 2013 [cit. 2014-07-13]. Dostupné z: <http://www.electronicaestudio.com/docs/istd012.pdf>
- [19] Fritzing. *Fritzing electronics* [online]. 2014 [cit. 2014-07-16]. Dostupné z: <http://fritzing.org/home/>

# Seznam příloh

Součástí BP je CD.

Adresářová struktura přiloženého CD:

- / Pdf - celý obsah bakalářské práce ve formátu PDF
- / Project - obsahuje projekt s naprogramovanou aplikací v Microsoft Visual studiu
- / App - obsahuje zdrojové soubory naprogramované aplikace a textový soubor readme.txt s pokyny na spuštění aplikace na Raspberry Pi
- / Testy - obsahuje soubory z naměřenými daty a fotografie pořízené během testování